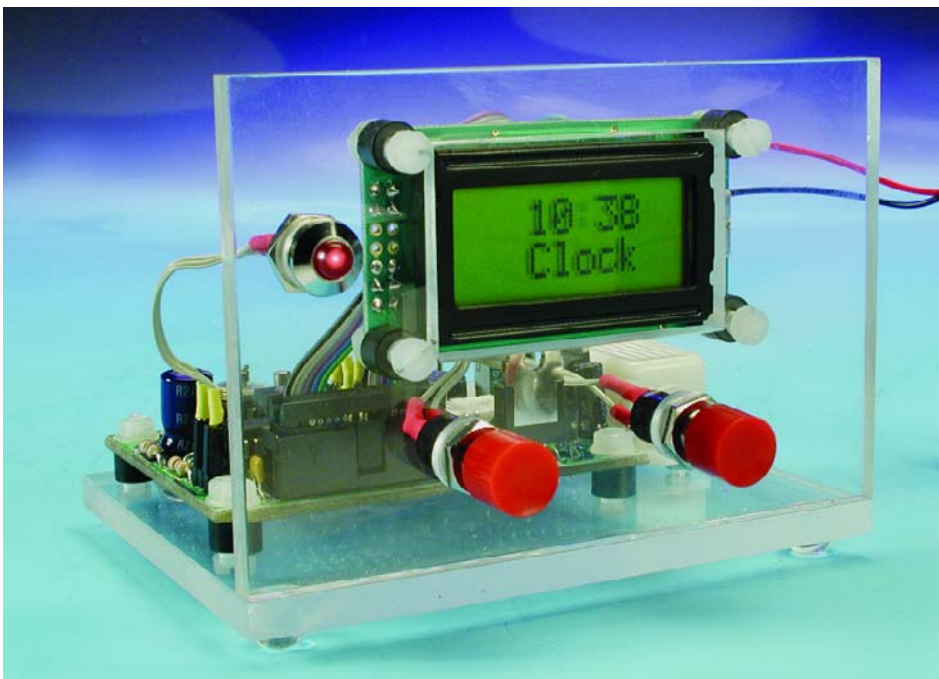


Horloge multi alarme

Plusieurs réveils en un seul

Kai-Uwe Mrkor

Un réveil ordinaire n'est pas exactement conçu pour attirer plusieurs fois l'attention au cours de la journée (par exemple : « il est temps de prendre vos pilules »). Il faut un dispositif qui garde plusieurs choses à « l'esprit ».



Le circuit de ce réveil multiple se distingue par sa simplicité. Il se compose, comme le montre immédiatement la **figure 1**, d'une combinaison microcontrôleur, composant horloge standard et afficheur LC. Le AT89C2051 de Atmel qui se trouve au cœur du circuit est un microcontrôleur 8 bits, compatible 8051, peu coûteux et très répandu. La liste de caractéristiques ci-dessous donne une idée des possibilités de ce petit contrôleur :

- compatibilité totale avec le « 51 » au niveau du logiciel
- 2 Koctets de mémoire Flash
- comparateur analogique intégré
- plusieurs sorties permettant d'attaquer une LED
- fonctionnement statique (de 0 à 24 MHz)
- extension de la plage de tension (de 2,7 à 6 V)

La petite taille du boîtier, qui ne comporte que 20 broches E/S, constitue le facteur déterminant dans notre application. Les ports 1 et 3 (hormis P3.6), donc 15 lignes de port en tout, sont programmables à volonté. C'est plus qu'il n'en faut pour raccorder la périphérie utilisée : un afficheur LCD pour l'heure et les menus, 2 poussoirs pour régler l'horloge et définir les heures d'alarme, une LED et un avertisseur sonore piézo.

Ces éléments E/S sont accompagnés d'un composant horloge DS1307 communiquant avec le contrôleur par bus I²C. Le DS1307 se charge de la mise à jour de l'heure (assurée par un élément lithium 3 V en cas de panne de courant) ; il fournit à la broche SOW/OUT un signal programmé au rythme d'une seconde.

L'erreur n'est pas qu'humaine

Un essai à cette broche indique toutefois dans la plupart des cas que la période dévie de la valeur promise (1,000 000 s). C'est la faute au quartz dont la valeur présente une certaine dispersion. L'auteur a mesuré un certain nombre de quartz et constaté que la période dévie jusqu'à 1,000 008 s. Un rien, me direz-vous.

Mais cette erreur de la sixième décimale après la virgule suffit à réduire considérablement la précision de l'horloge. Elle produit en effet une déviation d'environ 4 minutes par an. Le condensateur ajustable C5 permet de compenser la déviation. Il faut bien entendu disposer d'un compteur précis à la microseconde près... Raccorder simplement la sonde à la broche 7 du DS1307, puis régler le condensateur ajustable jusqu'à ce que la déviation se trouve dans des limites acceptables. Faute de compteur, il est préférable de ne pas monter C5.

Une alimentation équilibrée

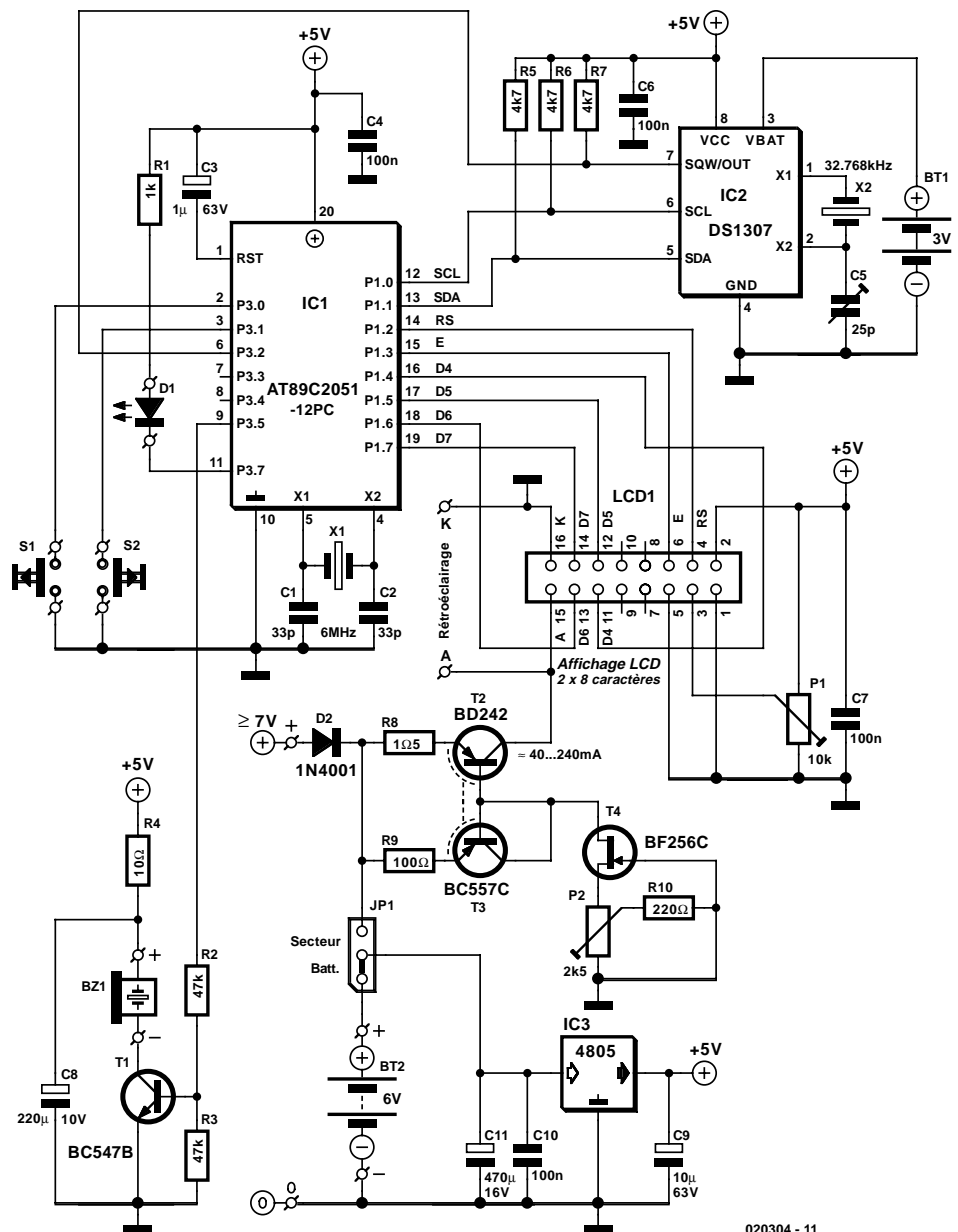
L'alimentation est quelque peu inhabituelle. Elle se compose d'un régulateur IC3 à faible chute de tension (*low-drop*) et de ses composants auxiliaires. Ce régulateur maintient la tension d'alimentation – provenant d'une pile ou d'une alimentation secteur – à 5 V pour tous les éléments du circuit, hormis le rétro-éclairage de l'afficheur LCD. Le cavalier JP1 empêche le rétro-éclairage, qui consomme entre 40 et 240 mA selon l'afficheur, de faire appel à la pile. Lorsqu'un bloc d'alimentation secteur est raccordé, par contre, la source de courant T2 à T4 ajustable par P2 (il s'agit de ce qu'on appelle en renfort et fournit le courant nécessaire au rétro-éclairage.

Le circuit, éclairage LCD non compris, consomme environ 5 mA. Il est possible d'abaisser légèrement cette valeur en réduisant quelque peu la fréquence d'horloge du microcontrôleur. La durée d'exécution des programmes du microcontrôleur n'est pas critique. Même des tests avec un quartz 2,457 6 MHz se sont déroulés sans aucun problème. La seule conséquence notable est une gestion quelque peu léthargique du menu de réglage de l'heure et des heures d'alarme.

Le DS1307 n'est bien entendu pas concerné car X2 lui fournit sa propre base de temps.

Logiciel

Le logiciel a été implémenté en C. L'environnement de développement



020304 - 11

Figure 1. La combinaison d'un contrôleur, d'un composant horloge, d'un afficheur LCD et de poussoirs de commande.

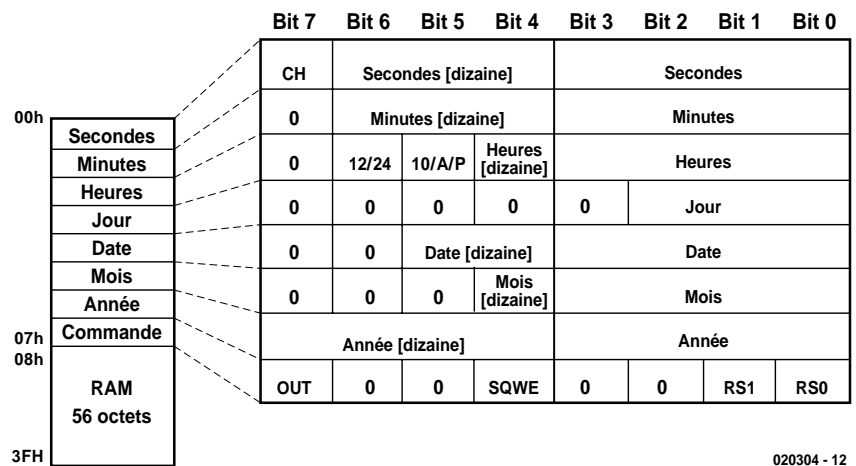
utilisé est μ Vision2 de l'entreprise Keil dont on peut obtenir sous www.keil.com/demo/eval/c51.htm une version d'évaluation gratuite. Les seules restrictions de cette version sont l'interdiction de l'utiliser à des fins commerciales et la limitation de la taille du code à 2 Koctets. Étant donné que le 89C2051 ne possède que 2 Koctets de mémoire Flash, cette limitation n'en est pas une ! Le programme du microcontrôleur gère l'affichage de l'heure, le menu de l'utilisateur, et émet un signal auditif lors de chacune des heures d'alarme.

Le rythme des secondes de l'horloge n'est pas généré par le contrôleur mais par le DS1307 et est transmis au contrôleur par une interface de bus I²C. Le AT89C2051 ne dispose pas d'une interface de bus I²C intégrée (contrairement à certains dérivés 8051 de Philips). Il faut donc implémenter en logiciel quelques modules I²C en s'inspirant d'un exemple de programme de Keil.

L'accès au composant horloge est similaire à celui d'une mémoire ordinaire, à la particularité près que les 8 premiers octets sont réservés au fonctionnement de l'horloge. La sauvegarde par pile permet d'utiliser sans crainte les 56 emplacements mémoire suivants pour les heures d'alarme.

Toutefois, avant que le DS1307 puisse fonctionner comme base de temps, il faut l'initialiser, c'est-à-dire le mettre à l'heure exacte et positionner les octets de commande lors de la mise en marche. Le schéma d'occupation de la mémoire de la **figure 2** indique la signification de chaque bit des 8 premiers octets. Le bit de poids fort de l'octet des secondes (CH, Clock Halt) joue un rôle particulièrement important. Il est toujours à 1 lorsque le circuit intégré est mis sous tension. Le composant horloge ne devient actif que lorsque CH est mis à 0. Le positionnement à 1 des bits OUT et SQWE de l'octet de commande indique au DS1307 d'envoyer un signal rectangulaire par la broche OUT. La fréquence de ce signal est déterminée par les bits RS1 et RS0. La combinaison 00b engendre un rythme de 1 s.

La liaison entre OUT et $\overline{\text{INT0}}$ du microcontrôleur engendre dès lors une interruption par seconde. Le sous-programme de traitement de ces interruptions met à jour la sortie de



020304 - 12

Figure 2. Occupation de la mémoire du DS1307.

l'heure avant de tester si l'heure présente correspond à celle d'une

alarme dans la RAM du DS1307. Si oui, 3 signaux acoustiques et

Liste des composants

Résistances :

R1 = 1 kΩ
 R2,R3 = 47 kΩ
 R4 = 10 Ω
 R5 à R7 = 4kΩ7
 R8 = 1Ω5
 R9 = 100 Ω
 R10 = 220 Ω
 P1 = ajustable 10 kΩ
 P2 = ajustable 2kΩ5

Condensateurs :

C1,C2 = 33 pF
 C3 = 1 μF/63 V radial
 C4,C6,C7,C10 = 100 nF
 C5 = 25 pF (ajustable)
 C8 = 220 μF/10 V radial
 C9 = 10 μF/63 V radial

C11 = 470 μF/16 V radial

Semi-conducteurs :

D1 = LED 5 mm rouge faible courant (avec fixation châssis le cas échéant)
 D2 = IN4001
 T1 = BC547B
 T2 = BD242
 T3 = BC557C
 T4 = BF256C
 IC1 = AT89C2051-I2PC (programmé **EPS 020304-41***)
 IC2 = DS1307 (Dallas Semiconductor)
 IC3 = 4805

Divers :

JP1 = embase à 1 rangée de 3 contacts + cavalier
 S1,S2 = bouton-poussoir unipolaire montage châssis

LCD1 = affichage LCD à 2 lignes de 8 caractères tel que, par exemple, AV0820 de Anag Vision) et embase à 2 rangées de 8 contacts à détrompeur
 X1 = quartz 6 MHz (résonance parallèle)
 X2 = quartz 32,768 kHz
 BT1 = pile-bouton 3 V au lithium telle que, par exemple, CR2032 + porte-pile encartable (diamètre 22,75 mm)
 BT2 = cf. texte
 BZ1 = résonateur CC 5 V/6 V
 Platine 020304-1*
 disquette 020304-11* (ou Téléchargement, cf. encadrés)

* Où les trouver :

www.elektor.fr/pcbs/pcbs.htm

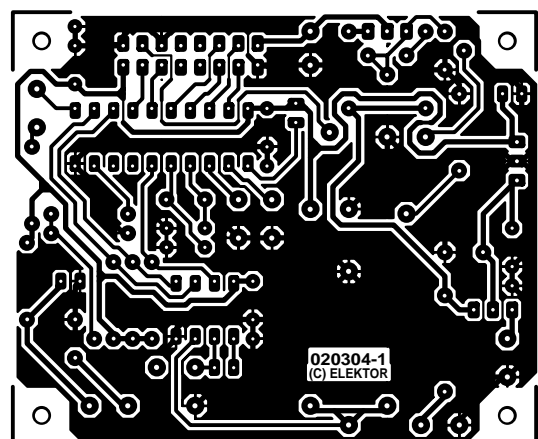
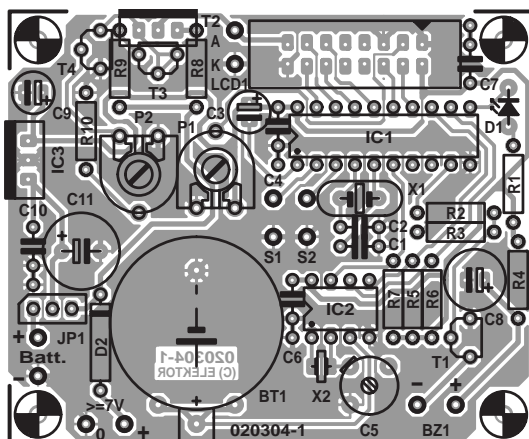


Figure 3. Disposition de la platine simple face.



optiques le signaleront. Le programme de commande teste ensuite si l'utilisateur a pressé un

des 2 poussoirs d'appel du menu. Si oui, le menu est traité, sinon le microprocesseur revient en mode

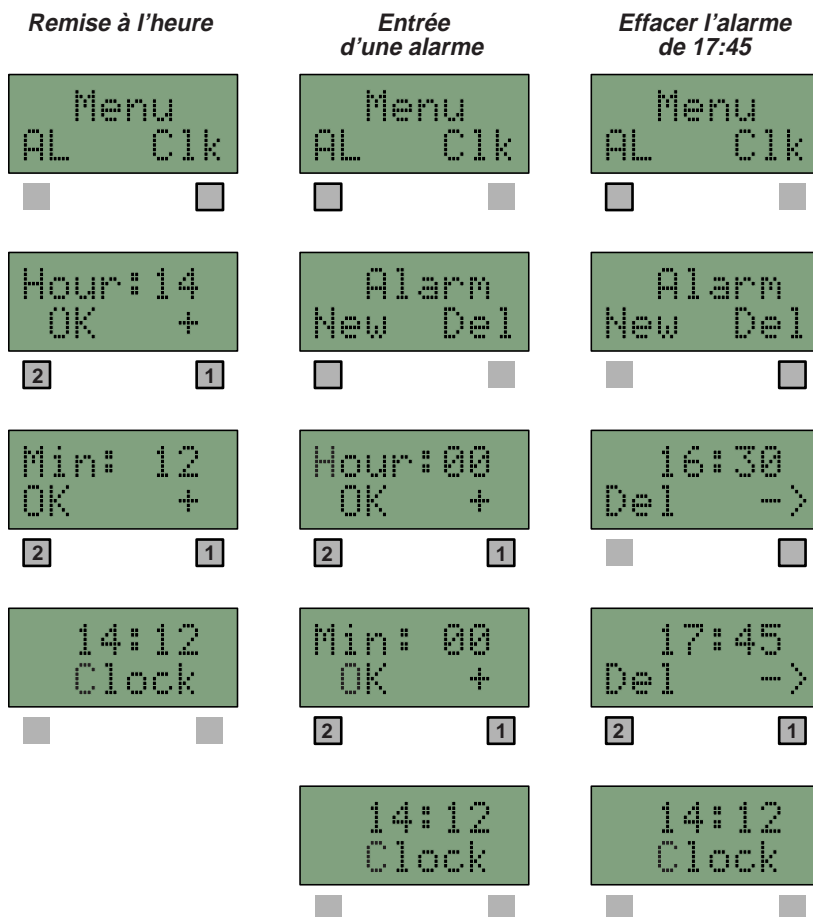
repos. Dans ce mode d'économie de courant, le AT89C2051 avec une fréquence d'horloge est de 6 MHz ne consomme que 1,5 mA au lieu de 7,5 mA en mode de fonctionnement normal. Le CPU ne se « réveille » que lors de l'interruption suivante du DS1307, et le cycle recommence.

Montage et fonctionnement

La disposition du circuit sur une platine simple face est reproduite dans la **figure 3**. Un certain nombre de composants ne se trouvent pas sur la platine : la « grosse » batterie BT2 (il peut s'agir de 2 éléments lithium, de 4 éléments alcalins ou d'une pile plate 9 V), le vibreur piézo BZ1, les 2 poussoirs S1 et S2 et, bien entendu, l'afficheur LC. Ce dernier est raccordé par un connecteur à 2-8 picots et 2 cosses à souder (A et K du rétro-éclairage). La pose des composants ne présente pas de problème, aucun d'entre eux ne requiert un traitement particulier. Il faut bien entendu connecter les condensateurs électrolytiques, les transistors et les circuits intégrés dans le bon sens. T2 et le régulateur de tension ne nécessitent pas de dissipateur thermique supplémentaire, vu la faible intensité des courants en jeu. Les 2 poussoirs sont fixés au boîtier, directement sous l'afficheur (S1 à gauche, S2 à droite), comme on peut le voir dans l'image du titre. La LED d'alarme peut être aussi montée sous l'afficheur.

Les 2 boutons-poussoirs permettent d'effectuer tous les réglages par menu. Presser longuement (≥ 1 seconde) un des 2 poussoirs pour entrer dans le menu. Le menu propose toujours un choix binaire. Les 2 possibilités apparaissent dans la seconde des 2 lignes de l'affichage LCD. Choisir l'option de menu désirée en pressant le poussoir situé au-dessous. Les 3 fonctions, « Mise à l'heure », « Introduire l'heure d'alarme » et « Supprimer l'heure d'alarme » représentées dans la **figure 4** se passent de commentaires particuliers.

(040302)



020304 - 13

Figure 4. Mode d'emploi « y a pas plus simple » du réveil multiple.

Téléchargements

Vous trouvez, à l'adresse www.elektor.fr/dl/dl.htm, les téléchargements suivants relatifs à ce projet :

- le programme du microcontrôleur en C (source et code hexadécimal)
- le dessin des pistes au format .pdf